Robust and Opportunistic Planning for Planetary Exploration

Daniel M. Gaines, Tara Estlin, Caroline Chouinard, Forest Fisher
Rebecca Castaño, Robert C. Anderson, and Michele Judd

Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Dr., Pasadena CA 91109
{*firstname.lastname*}@jpl.nasa.gov

Abstract

Planning for rover operations involves a significant amount of uncertainty. With limited a priori knowledge of the area a rover will explore, it is difficult to predict the affects of actions including their duration and the amount of resources they will consume. In addition, the system may not even know ahead of time all of the goals it will be asked to achieve as new opportunities may be identified during the mission. We are developing the OASIS system to enable rovers to generate and execute high quality mission operations plans and to identify and exploit new science opportunities that may arise during the mission. OASIS combines planning and machine learning techniques to achieve these results. In this paper we discuss how OASIS handles these types of uncertainties and present results from testing the system in simulation and on rover hardware.

1 Introduction

Planetary exploration by its nature involves a significant amount of uncertainty. The objective of such missions is to gather information about previously unknown areas. As such, little a priori information may be available about the nature of the terrain a rover must explore or the obstacles that it will encounter. This makes it challenging to develop an operation sequence as it is difficult to estimate the time and resources required by rover activities.

In addition, we are developing technologies that enable rovers to identify potentially interesting science opportunities on their own. This will provide important capabilities for rovers such as enabling rovers to identify opportunities that might have otherwise gone unnoticed or to take advantage of short-lived science opportunities such as a passing dust devil. However, this capability also adds another element of uncertainty to mission operations as the rover will not know ahead of time all the science goals it will be asked to work on. New goals with different priorities may be posted to the system at any time.

We have developed the OASIS (Onboard Autonomous Science Investigation System) integrated science analysis and planning system that enables planetary rovers to generate and execute high quality mission operations plans in the presence of these types of uncertainty. OASIS includes a continuous planning system to generate operations plans given prioritized science goals and mission constraints and to monitor and repair plans during execution. The system also includes a data analysis unit that uses machine learning algorithms to perform onboard processing of collected science data. When a science opportunity is detected, one or more requests are sent to the planning and execution system which attempts to accomplish these additional objectives while still achieving current mission goals.

2 OASIS

The OASIS system provides onboard science analysis coupled with planning and execution. The system enables a rover to carry out prioritized science goals commanded from Earth as well as opportunistic science goals identified by onboard data analysis. Figure 1 shows the main components of the OASIS system and how they interact to analyze data and re-task the rover to respond to opportunistic science events. OASIS consists of the following components:

Planning and Scheduling: generates operations plans for mission goals and dynamically modifies plan in response to new science requests.

Execution: carries out the rover functional capabilities to perform the plan and collect data. Oasis TDL [Simmons and Apfelbaum, 1998] for its Executive and the CLARAty [Nesnas *et al.*, 2003] functional layer for low-level robotic capabilities.

Feature Extraction: detects rocks in images and extracts rock properties (e.g. shape and texture).

Data Analysis: uses extracted features to assess the scientific value of the planetary scene and to generate new science objectives that will further contribute to this assessment.

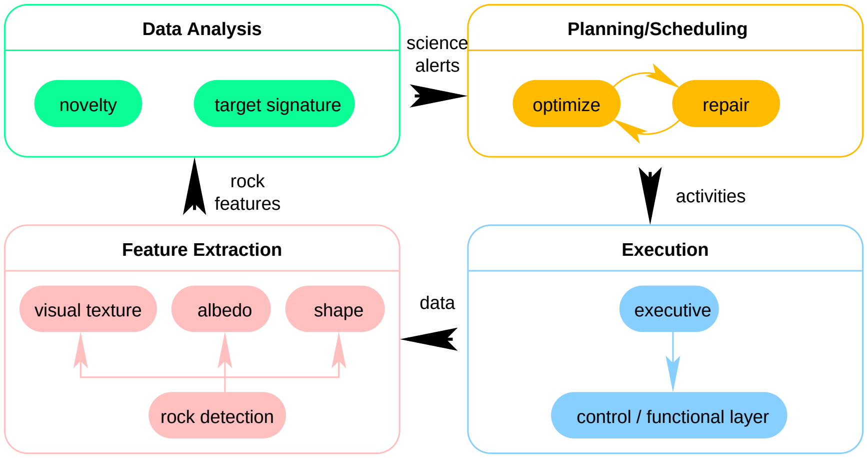

Figure 1: OASIS architecture.

The feature extraction and data analysis components of OASIS have been described previously in [Castano *et al.*, 2004]. Here we will give a brief overview of these components and concentrate on the planning and scheduling unit and how it supports opportunistic science.

2.1 Feature Extraction

Our initial emphasis in OASIS has focused on image analysis and the characterization of surface rocks. Rocks are among the primary features populating the Martian landscape and the understanding of rocks on the surface is a first step leading to more complex regional geological assessments.

Images are segmented using a rock detection algorithm based on edge detection and tracing. Next, a set of properties is extracted from each rock. Our feature extraction priorities are based upon our knowledge of how a geologist in the field would extract information. Important features to look for and categorize include albedo (an indicator of rock surface reflectance properties), visual texture (which provides valuable clues to mineral composition and geological history), shape, size, color and arrangement of rocks. Currently our system identifies the first three of this set; future work will expand this to cover additional features.

2.2 Data Analysis

After features have been extracted from each rock, OASIS runs a set of data analysis algorithms to look for interesting rocks. Two of these algorithms can result in the generation of science alerts: key target signature and novelty detection.

Key Target Signature: enables scientists to efficiently and easily stipulate the value and importance of certain features. Scientists often have an idea of what they expect to find during a rover mission and/or are looking for specific clues that reflect signs of life or water (past or present). Using this technique, target feature vectors can be pre-specified and an importance value assigned to each of the features. Rocks are then prioritized as a function of the weighted Euclidean distance of their extracted features from the target feature vector.

Novelty Detection: detects and prioritizes unusual rocks that are dissimilar to previous rocks encountered. We have looked at three different learning techniques for novelty detection: distance-based using k-means clustering, probability-based using Gaussian mixture models and discrimination-based using kernel one-class classifier. The general idea is that as the rover collects data about the rocks in an area, the machine learning techniques will enable it to build a model of the characteristic rocks. If a new sample falls well outside of this model, then it is considered novel and potentially worthy of further investigation.

2.3 Science Alert Protocol

Using the above algorithms, the data analysis software can flag rocks that should be further analyzed and produce a new set of measurement goals. We call this capability the *science alert*, since it alerts other onboard software that new and high priority science opportunities have been detected. OASIS currently supports two types of alerts. A *stop and call home* alert indicates that the rover should remain at its current location until it has received further instructions form Earth. Such an alert would typically be reserved for situations in which data analysis has made an extremely interesting observation and the rover should stay where it is to avoid the risk of losing the target. The second class of alerts is *data sample requests* in which the rover is requested to perform an additional science measurement and then continue on with previously scheduled activities. Achieving this alert may require the rover to change its heading or possibly its position.

2.4 Planning and Scheduling

The objectives of OASIS's planning and scheduling component is to maximize the value of the science that is performed by the rover and to ensure that the operations plan satisfies rover and mission constraints. To provide robust execution, the system must respond to problems that might arise during plan execution, such as an activity consuming more resource than expected. To maximize the value of the plan, the system must exploit opportunities that arise. These may include additional available time due to an activity taking less time than expected or a new, highly interesting goal that has been identified by Science Analysis.

Planning and scheduling capabilities in OASIS are provided by CASPER [Estlin *et al.*, 2002; Chien *et al.*, 2000], which employs a continuous planning technique where the planner continually evaluates the current plan and modifies it when necessary based on new state and resource information. Rather than consider planning a batch process, where planning is performed once for a certain time period and set of goals, the planner has a current goal set, a current rover state, and state projections into the future for that plan. At any time an incremental update to the goals or current state may update the current plan. This update may be an unexpected event (such as a new science opportunity) or a current reading for a particular resource level (such as power). The planner is then responsible for maintaining a plan consistent with the most current information.

A plan consists of a set of grounded (i.e., time-tagged) activities that represent different rover actions and behaviors. Rover state in CASPER is modeled by a set of plan timelines, which contain information on states, such as rover position, and resources, such as power. Timelines are calculated by reasoning about activity effects and represent the past, current and expected state of the rover over time. As time progresses, the actual state of the rover drifts from the state expected by the timelines, reflecting changes in the world. If an update results in a problem, such as an activity consuming more memory than expected and thereby over-subscribing RAM, CASPER re-plans, using iterative repair [Zweben *et al.*, 1994], to address conflict.

CASPER includes an optimization framework for reasoning about soft constraints. User-defined preferences are used to compute plan quality based on how well the plan satisfies these constraints. Optimization proceeds similar to iterative repair. For each preference, an optimization heuristic generates modifications that could potentially improve the plan score.

We have developed a domain specific control algorithm within CASPER to support the objectives of maximizing plan quality and ensuring robust execution. Figure 2 provides a high level description of this algorithm.

Initial Plan Generation

We use a Depth First Branch and Bound algorithm to generate the initial operations sequence. The input to the system is a set of prioritized science requests and constraints on the time and energy available for carrying out the mission. The initial plan maximizes the value of science goals that can be achieved under these constraints.

Plan Execution

CASPER monitors updates from the Executive as the plan is executed, checking for problems that must be resolved or opportunities that can be exploited. A problem can occur with an activity at any point during its lifetime. For examples, an update may indicate that there will be a problem with an activity scheduled to start at some time in the future. In this case, CASPER will use iterative repair as part of the optimization loop to try to resolve the conflict.

Problems may also occur for activities that have already been passed to the Executive but have not yet begun execution. In this case, CASPER will send a rescind message for the problematic activity to the Executive. If the Executive receives the message before the activity has begun execution, it will delete it and send CASPER a confirmation. If the activity has already begun execution, the Executive will abort the activity and send an update to CASPER once the activity has been aborted.

The Executive itself monitors problems with activities that are currently executing. If a problem is detected, it is the responsibility of the executive to abort the activity and send an update to CASPER to let CASPER know that the activity was aborted.

While the first priority of the planning and scheduling system is to ensure robust execution, it is also continually checking for opportunities to increase the value the mission. An update from the Executive may indicate that an activity took less time or energy than predicted. In this case, it may be possible to achieve a goal that was not included in the initial plan. During the optimization loop, if all conflicts have been resolved, CASPER will select a high priority goal from the set of unsatisfied goals and add it to the schedule. This will most likely introduce new conflicts and the following optimization iterations will be spent trying to resolve them. If the conflicts can be resolved, the plan score will be increased and this plan will be saved as the best seen so far.

If an opportunistic science opportunity has been identified by Data Analysis, CASPER will try to add it to the plan. Again, this is likely to introduce conflicts and iterative repair will be used to try to fix them. It may be that the rover's schedule is too constrained to achieve the opportunistic goal. We set a timer for each opportunistic goal and if the timer expires before the goal is achieved, the goal is permanently deleted.

As a final check to try to maximize the use of rover resources, after the optimize loop, if there are no currently executing activities, CASPER will look ahead in the schedule to see if a future activity can be moved up in time without causing a conflict. If so, this will result in packing the schedule, limiting rover idle time.

Input

Prioritized science goals from Earth
Time constraint
Resource constraints

Initial Plan Generation

Run Depth First Branch and Bound given initial science goals and constraints

Plan Execution

While running
- Get current time
- Process any updates from Executive
- For each activity scheduled to start within <n> seconds
  - If activity does not contribute to an existing conflict, send to Executive
- If there are conflicts in the schedule
  - If an activity already sent to executive is contributing, rescind activity
- **Optimize:**
  - for i = 1 to num_optimize_iterations
    - If score of current plan is best so far, save plan
    - If there is an unsatisfied opportunistic science goal, satisfy it
    - Else, if there are conflicts, perform an iteration of repair
    - Else, if there are unsatisfied science goals, satisfy one from set of highest priority science goals
  - Reload plan with highest score
- If an opportunistic science goal has not been satisfied for opsci_time_limit, delete the goal
- If no activities are currently executing, check if an activity in the future can be moved up in time

Figure 2: CASPER control algorithm for rover domain.

3 System Testing

To evaluate our system we performed a series of tests both in simulation and using rover hardware in the JPL Mars Yard (Figure 3). These tests covered a wide range of scenarios that included the handling of multiple, prioritized science targets, limited time and resources, opportunistic science events, resource usage uncertainty causing under or over-subscriptions of power and memory, large variations in traverse time, and unexpected obstacles blocking the rover's path.

Our testing scenarios typically consisted of a number of science targets specified at certain locations. A map was used that would represent a sample mission-site location where data would be gathered using multiple instruments at a number of locations. Figure 4 shows a sample scenario that was run as part of these tests. This particular map is of the JPL Mars Yard. The pre-specified science targets represented targets that would be communicated by scientists on Earth. These targets were typically prioritized and for many scenarios constraints on time, power or memory would limit the number of science targets that could be handled. The map also shows the path that was planned for the rover and the path the rover actually followed. These are not necessarily the same as the planned path does not account for all the obstacles the rover may have to avoid. A large focus of our tests was to improve system robustness and flexibility in a realistic environment. Towards that goal we used a variety of target locations and consistently selected new science targets and/or new science target combinations that had not been previously tested.

Another primary scenario element was dynamically identifying and handling opportunistic science events. For these tests, we concentrated on a particular type of event, which was finding rocks with a high albedo measurement (i.e., light or white-colored rocks). This setting was an example of using the data analysis algorithm for target signature, where a particular terrain signature is identified as having a high interest level. If rocks were identified in hazard camera imagery that had a certain interest score, then a science alert was created and sent to the planner. If a science alert was detected the planner attempted to modify the plan so an additional image of the rock of interest was acquired.

Other important scenario elements included adding or deleted ground-specified science targets based in resource under or over-subscriptions. For instance, in some tests, the rover covered distances faster than expected and the planner was able to add in additional science targets that could not be fit into the original plan. Conversely, in other tests, the rover used more power than expected

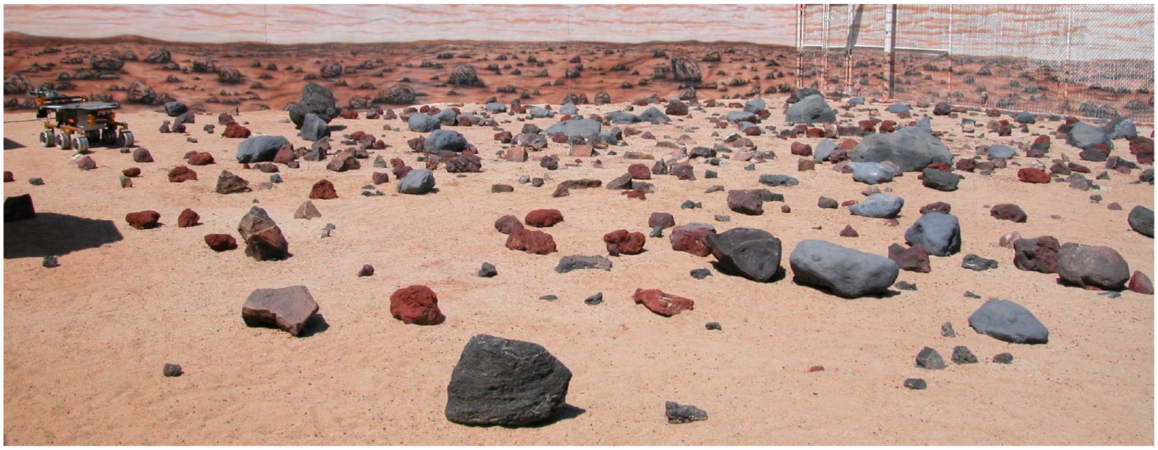

Figure 3: Testing with the FIDO rover in the Mars Yard.

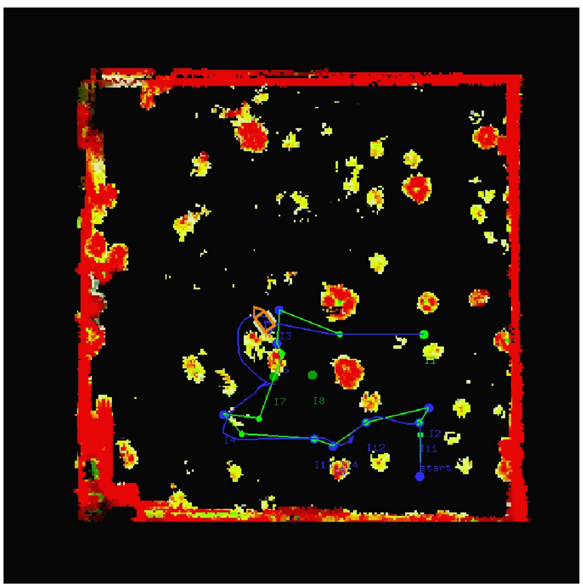

Figure 4: Example scenario.

during traverses (or science measurements), which eventually caused a power over-subscription. The planner resolved this situation by deleting some lower priority science targets. Unexpected energy drops during a traverse could also be handled by the executive, which detects the shortfall and stops the current traverse if there is not enough energy to complete it. In all cases, the planning and execution system attempts to preserve as many high priority science targets as possible with current resource and time settings.

3.1 Discussion of Test Results

We are in the process of developing a formal evaluation process by which we will be able to obtain quantitative measurements of how well our system provides robust and opportunistic planning and execution. At this point we have more anecdotal results from our extensive testing in simulation and with rover hardware in the JPL Mars Yard.

Tests in the Mars Yard typically consisted of 20-50 meter runs over a 100 square meter area with many obstacles that cause deviations in the rover's path. Most rocks in the Mars Yard are dark in color, thus we brought in a number of whiter rocks to trigger science alerts during rover traverses. Science measurements using rover hardware were always images, since other instruments were not readily available (e.g., spectrometer). However different types of measurements were included when testing in simulation.

As a final test of our system, we performed a several hour long demonstration in October 2004. This demonstration covered the elements previously presented in this section. Further, the combination of science targets used had not been previously tested with. This set also included a science target that was selected that day by a present Mars Exploration Rover (MER) scientist. Rocks intended to cause science alerts were also placed in new locations not previously used. Overall, the demonstration was very successful. Two scenario runs were performed. Both had multiple targets with time or resource constraints preventing all targets from being included in the initial plan. In the first run a number of science alerts were correctly identified and handled. This run also had an additional science target added dynamically in the run due to the rover traveling faster than estimated. In the second run, lower priority targets were deleted due to more power being used in early traverses than expected. The software presented in this paper (planning, scheduling, execution, feature extraction and data analysis) operated correctly in all cases and caused no undesirable behavior. In general, the rovers operated fully autonomously and traveled over 40 meters.

While the system performed well during testing, we have identified some areas for in which the system's handling of uncertainty could be improved. While the planning system can respond appropriately when activities do not run in the estimated time (whether they take more time or less time than predicted) it would be better if the system could make more accurate predictions as the planner could do a better job optimizing the value of the mission plan. This would reduce the time the planner spends replanning and, in some cases, could result in higher quality plans. As an example of how this could result in higher quality plans, consider the case where activities take less time than the planner predicts. If the planner overestimated the duration of activities, it might leave out a low priority science goal near its starting position because it estimated there would be insufficient time to accomplish the goal. However, by the end of the plan, the planner finds it had more time than expected and can now include additional goals. However there may be insufficient time at this point to go back to that earlier goal. If the planner had a more accurate prediction of activity durations it would have known to include the goal in its schedule and would have completed it earlier on.

The challenge in making such predictions is that the duration of traverse activities depend on the nature of the terrain and the amount of obstacles the rover will encounter, which can be difficult to predict ahead of time. A possible solution may be to allow the rover adjust its predictive model of its activities based on its experience during mission. Techniques such as regression tree learning have been shown to allow robots to learn such predictive models for navigation actions. CITE

Another improvement would be to explicitly reason about the uncertainty of activities. This would enable the planner to make tradeoffs between actions that may result in the collection of valuable science but may have a high uncertainty in the outcome.

4 Related Work

The objectives of OASIS are similar to those of the Autonomous Sciencecraft Experiment (ASE) [Sherwood *et al.*, 2003] which also uses science analysis to generate additional goals for a planner. OASIS differs from ASE in the types of feature extraction and data analysis that are performed. In addition, while ASE has focused on planning for orbiter missions, the focus for OASIS has been on ground operations. To support this type of planning OASIS must deal with the high degree of uncertainty inherent in ground operations and integrate path planning into the planning and scheduling process. Finally, in OASIS it is often necessary to temporarily halt currently executing activities, such as a traverse, in order to accomplish new science goals.

A number of other systems have used planning methods to coordinate robot behavior (e.g. [Bonasso *et al.*, 1997; Alami *et al.*, 1998]). However, these systems generate plans with a batch approach where plans are generated for a certain time period and if re-planning is required, an entire new plan must be produced. In OASIS, plans are continuously modified in response to changing conditions and goals. The CPS planner generates contingent plans which are then executed onboard a rover and can be modified at certain points if failures occur [Bresina *et al.*, 1999]. Since only a limited number of contingencies can be anticipated, our approach provides more onboard flexibility to new situations. If a situation occurs onboard for which there is not a pre-planned contingency, the rover must be halted to wait for communication with ground.

5 Conclusions

OASIS supports opportunistic science by integrating data analysis algorithms, which identifies potentially interesting science measurements, with planning and scheduling algorithms, which enables the rover to respond to these new requests. Our current system has been tested with several scenarios in simulation and on prototype rover hardware. In these scenarios we demonstrate the systems ability to respond appropriately to problems with plan execution and to exploit unexpected opportunities that might arise.

Currently, the planner preserves the original mission goals when attempting to perform opportunistic science. We will relax this constraint and allow the system to use priorities to determine when it is appropriate to achieve opportunistic science at the cost of existing goals. There are significant challenges with introducing autonomous techniques into the mission operations culture. We are taking steps to address this by introducing MER scientists to off-line versions of our software.

Acknowledgments

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. This work was funded by the NASA Intelligent Systems program and the JPL Interplanetary Network Directorate. We wish to thank the other members of the OASIS team for their work on feature extraction and data analysis and the CLARAty team, especially Issa Nesnas and Max Bajracharya, for their assistance with the rover.

References

[Alami *et al.*, 1998] R. Alami, R. Chautila, S. Fleury, M. Ghallab, and F. Ingrand. An architecture for autonomy. *International Journal of Robotics Research*, 17(4), April 1998.

[Bonasso *et al.*, 1997] R. Bonasso, R. Firby, E. Gat, D. Kortenkamp, D. Miller, and M. Slack. Experiences with an architecture for intelligent, reactive agents. *Journal of Experimental and Theoretical Artificial Intelligence Research*, 9(1), 1997.

[Bresina *et al.*, 1999] J. Bresina, K. Golden, D. E. Smith, and R. Washington. Increased flexibility and robustness for Mars rovers. In *Proceedings of the*

Fifth International Symposium on Artificial Intelligence, Robotics, and Automation in Space, Noordwijk, Netherlands, 1999.

[Castano *et al.*, 2004] R. Castano, M. Judd, T. Estlin, R. Anderson, L. Scharenbroich, L. Song, D. Gaines, F. Fisher, D. Mazzoni, and A. Castano. Autonomous onboard traverse science system. In *IEEE Aerospace Conference*, Big Sky, Montana, March 2004.

[Chien *et al.*, 2000] Steve Chien, Russell Knight, Andre Stechert, Rob Sherwood, and Gregg Rabideau. Using iterative repair to improve the responsiveness of planning and scheduling. In *Fifth International Conference on Artificial Intelligence Planning and Scheduling*, Breckenridge, CO, April 2000.

[Estlin *et al.*, 2002] Tara Estlin, Forest Fisher, Daniel Gaines, Caroline Chouinard, Steve Schaffer, and Issa Nesnas. Continuous planning and execution for a mars rover. In *Third International NASA Workshop on Planning and Scheduling for Space*, Houston, TX, October 2002.

[Nesnas *et al.*, 2003] Issa A. Nesnas, Ann Wright, Max Bajracharya, Reid Simmons, Tara Estlin, and Won Soo Kim. Claraty: An architecture for reusable robotic software. In *Proceedings of SPIE Aerosense Conference*, Orlando, Florida, 2003.

[Sherwood *et al.*, 2003] Robert Sherwood, Steve Chien, Rebecca Castano, and Gregg Rabideau. The autonomous sciencecraft experiment. In *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, March 2003.

[Simmons and Apfelbaum, 1998] Reid Simmons and David Apfelbaum. A task description language for robot control. In *Proceedings of Conference on Intelligent Robotics and Systems*, Vancouver Canada, October 1998.

[Zweben *et al.*, 1994] M. Zweben, B. Daun, E. Davis, and M. Deale. Scheduling and rescheduling with iterative repair. In M. Fox and M. Zweben, editors, *Intelligent Scheduling*, pages 241–256. Morgan Kaufmann Publishers Inc., 1994.